IN4393-16 Computer Vision - Q3
Assignment 1
Linear Filters: Gaussians and Derivatives

February 15, 2018

Students should work on this assignment with group of two people. The assignment must be presented by the group at lab sessions and it is due on 26th February 2018.

In this assignment, you are working with Gaussian filters and applying them to images, exploiting their separability property.

1   1D Gaussian Filter

In this first part, you will be implementing a 1D discrete Gaussian filter:

$$G_\sigma = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/2\sigma^2} \tag{1}$$

where σ is the variance of the Gaussian. The Matlab fucntion should look like this:

```
1  function G = gaussian( sigma )
2  ...
3  end
```

2   Convolving an image with a 2D Gaussian

You will write a function to convolve an image with a 2D Gaussian. One of the most important properties of Gaussian kernels is separability. Therefore,

convolving an image with a 2D Gaussian is equivalent to convolving the image along the x-axis and the y-axis separately with a 1D Gaussian. The Matlab function should use the previous function and should look like this:

```
1  function imOut = gaussianConv(image_path , sigma_x , sigma_y)
2  ...
3  end
```

Hint: use Matlab function conv2.

3   Comparing with Matlab's Gaussian Filter

Matlab already has a built-in function implementing 2D Gaussian kernels. The Matlab code for using it is:

```
1  G=fspecial('gaussian',n,sigma)
2  imOut=conv2(imIn,G,'same')
```

In this part, you can compare your implementation of 2D Gaussian convolution, through two 1D operations, to Matlab's. You can read any image and check the difference between your output image and Matlab's.

4   Gaussian Derivative

The Gaussian first order derivative is given by:

$$
\begin{aligned}
\frac{d}{dx}G_\sigma &= \frac{d}{dx}\left(\frac{1}{\sigma\sqrt{2\pi}}e^{-x^2/2\sigma^2}\right) \\
&= -\frac{1}{\sigma^3\sqrt{2\pi}}\,x\,e^{-x^2/2\sigma^2} \\
&= -\frac{x}{\sigma^2}G_\sigma
\end{aligned}
$$

Write a function to implement this first order derivative. Your function should look like this:

```
1  function Gd =gaussianDer(G , sigma)
2  ...
3  end
```

5   Gradient Magnitude and Orientation

Write a function that returns two images with the magnitude and orientation of the gradient for each pixel of the input image. Your function should look like this:

```
1  function [magnitude , orientation] = gradmag(img , sigma)
2  ...
3  end
```

5.1

Visualize your result using:

```
1  imshow(orientation , [-pi,pi])
2  colormap(hsv);
3  colorbar;
```

Visualize your result using Matlab quiver function.

5.2

Get results for varying sigma values.

1. How do the magnitude images change as the sigma increases?

2. How do the orientation images change as the sigma increases?

5.3

It is typical to place a threshold on the gradient magnitude so that the edge detection result is binary. Use different thresholds on the gradient magnitude and display your thresholded image for varying sigma values.

5.4

The Gaussian second order derivative is given by:

$$\frac{d^2}{dx^2}G_\sigma = \frac{d^2}{dx^2}\left(\frac{1}{\sigma\sqrt{2\pi}}e^{-x^2/2\sigma^2}\right)$$
$$= \frac{1}{\sigma^5\sqrt{2\pi}}\left(-\sigma^2 + x^2\right)e^{-x^2/2\sigma^2}$$
$$= \frac{-\sigma^2 + x^2}{\sigma^4}G_\sigma$$

Write a function that give an output of the specified derivative of an input image (type : 'x', 'y', 'xx', 'yy', 'xy', 'yx'). Your function should look like this:

```
1  function F=ImageDerivatives(img , sigma , type)
2  ...
3  end
```

5.5

Create an impulse image (all zeros, with only one pixel in the center set to the maximum value) and convolve the impulse image with 0, 1st, and 2nd order gaussian derivatives, try some sigma values and visualize the result images. Describe what you see. Is there a relation between the order and the sigma?